CHE221

SIMULATION LAB 7

-PRAGATI PATEL (230765) UG CHE

Aim of The Report:

1. Write a MATLAB function to compute the volume of a Van der Waals gas at fixed P and T using the fzero() function.
2. Plot the fugacity coefficient as a function of pressure (1 - 50 bar) for different mole fractions of component a.
3. Convert the MATLAB code to a function and integrate it with fugacity_slide.m to enable dynamic variation of the interaction parameter using a slider.

Introductory Theory:

Fugacity in Mixtures:

Fugacity of a component *a* in a binary mixture of *a* and *b* is given by

$$RT \ln\left(\frac{\hat{f}_a^v}{y_a P}\right) = -\int_V^\infty \left(\frac{\partial P}{\partial n_a}\right)_{V,T,n_b} dV.$$

For the Van der Waals equation of state, this integral simplifies to

$$\ln \hat{\phi}_a^v = -\ln\left(\frac{P(v-b_{mix})}{RT}\right) + \frac{b_a}{v-b_{mix}} - \frac{2y_a a_a + y_b\sqrt{a_a a_b}}{RTv}.$$

Where
v is the total molar volume
ya, yb are the mole fraction of a and b
ba, bb are excluded molar volumes aa,ab are interaction parameters
bmix= ∑yibi
Given Data: • aa = 0.2Jm³ mol⁻², • ab =0.3 Jm³ mol⁻² • ba = 0.00004 m³ mol⁻², • bb= 0.00003m³ mol⁻²

Simulation Methodology:

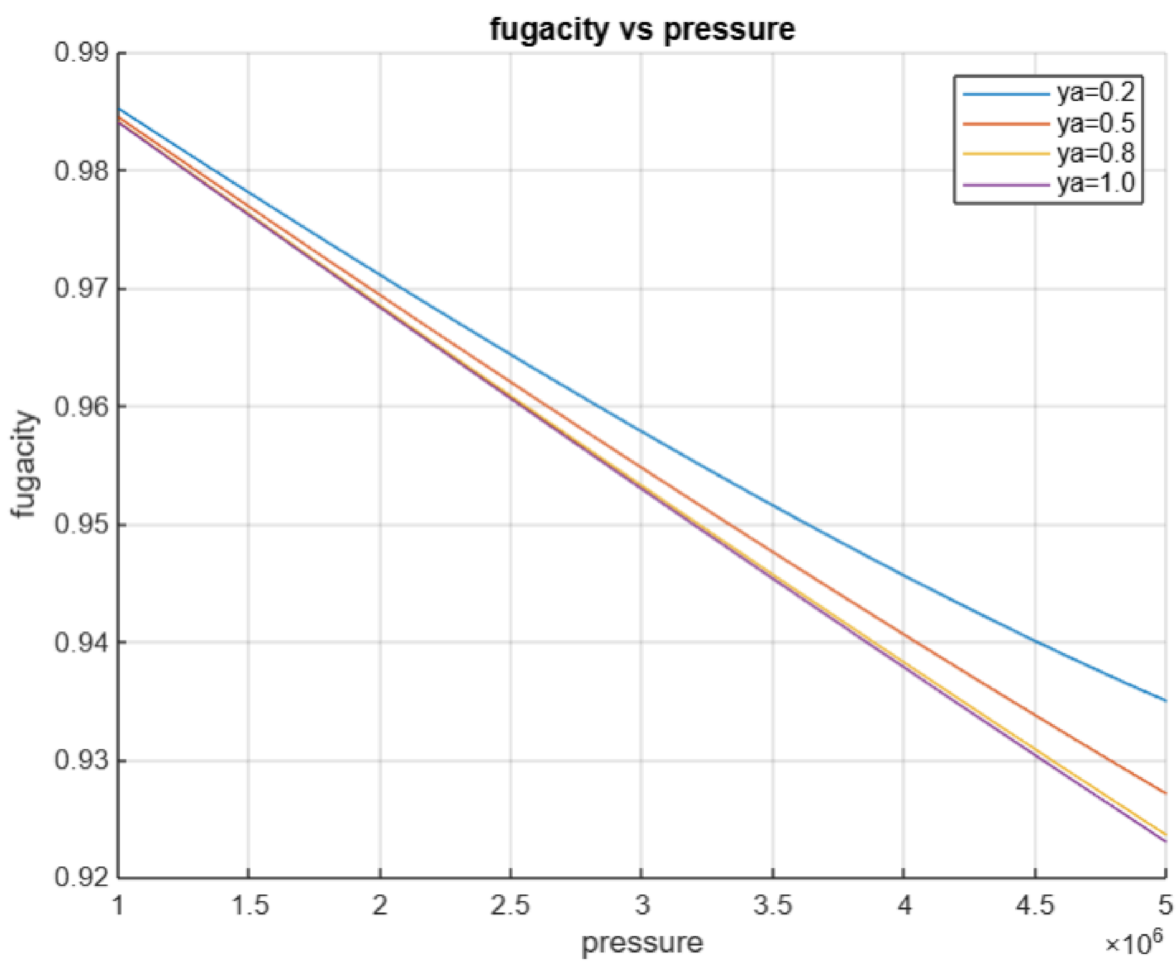
1. Computing Volume Using Van der Waals EOS

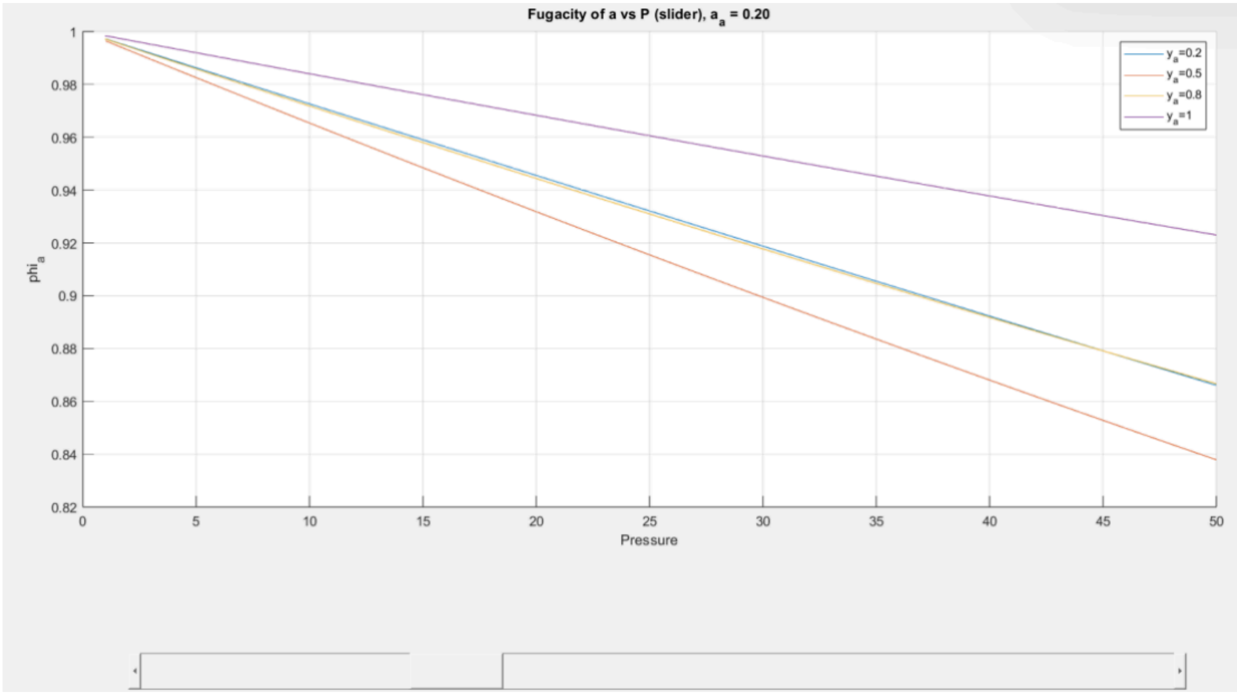
• Define a residual function for Van der Waals EOS.
• Use MATLAB's fzero() to solve for v at given P and T.
• Compute for pressures ranging from 1-50 bar.
• Repeat for different mole fractions .
• Plot vs. pressure.
• Convert the MATLAB script into a function.
• Modify fugacity_slide.m to allow real-time adjustment of.
• Observe its effect on fugacity coefficient plots.

Results & Discussion:

1.Volume Computation: The fzero() function successfully determined v for different pressures which is used for calculating phi.

2. Fugacity Coefficient Variation:

- The plot showed that decreases with increasing pressure.
- As increases, deviates more from ideal behavior.
- The Van der Waals model captures non-idealities but has some limitations at high pressures.
- 

3. Dynamic Adjustment Using Slider:
- Changing affected fugacity significantly.
- Higher values led to stronger intermolecular forces and lower fugacity coefficients.
- 

Appendix:

```
clc
close all
clear all
a_a=0.2;
a_b=0.3;
b_a=0.00004;
b_b=0.00003;
T=300;

y_a=[0.2 0.5 0.8 1.0];
P_low=10e5;
P_high=50*1e5;
P=linspace(P_low,P_high,50);
```

```
figure
hold on
for i=1:length(y_a)

y_b=1-y_a(i);
a_mix=y_a(i)^2*a_a+2*y_a(i)*y_b*sqrt(a_a*a_b)+y_b^2*a_b;
b_mix=y_a(i)*b_a+y_b*b_b;
v_lower=b_mix+1e-10;
v_higher=100000;
%V=zeros(length(P),1);
phi=zeros(length(P),1);
for j=1:length(P)
v=fzero(@(v) vol(a_mix,b_mix,P(j),T,v),[v_lower,v_higher]);
%V(i)=v;
phi(j)=-log((P(j)*(v-b_mix))/(8.314*T))+b_a/(v-b_mix)-2*(y_a(i)*a_a+y_b*((
a_a*a_b)^0.5))/(8.314*T*v);
end
plot(P,exp(phi));

end
hold off
grid on
legend('ya=0.2','ya=0.5','ya=0.8','ya=1.0')
title('fugacity vs pressure');
xlabel('pressure');
ylabel('fugacity');
function  volume=vol(a,b,P,T,v)
volume=(P+a/(v*v))*(v-b)-8.314*T;

end
```

```
clear
a_a=0.2;
fugacity_slider

function plot_fugacity_mixture(a_a,ax)
hold(ax,'on');
```

```
a_b = 0.3 ;
b_a = 0.00004 ;
b_b = 0.00003;
y_aii=[0.2,0.5,0.8,1];

T=300;%K
R=8.314;
for j=1:4
 y_a=y_aii(j);
 y_b=1-y_a;
 a_mix=y_a^2*a_a + 2*y_a*y_b*sqrt(a_a*b_b)+y_b^2*a_b;
b_mix=y_a*b_a+y_b*b_b;
 phi_a=zeros(1,50);

for i=1:50

 myfun = @(v,P) (P+a_mix/(v*v))*(v-b_mix)-R*T;
 P=i*10^5;
fun = @(v) myfun(v,P);
vo=[b_mix+1e-10,100000];
v = fzero(fun,vo);
phi_a(i)=
exp(-log(P*(v-b_mix)/(R*T))+b_a/(v-b_mix)-2*(y_a*a_a+y_b*sqrt(a_a*a_b))/(R
*T*v));
end
disp(phi_a)
figure(1)
P__=linspace(1,50,50);
plot(P__,phi_a,'-');
xlabel('Pressure');
ylabel('phi_a');
legend('y_a=0.2','y_a=0.5','y_a=0.8','y_a=1');
title('φ_a as a function of pressure');
grid on
hold on
end

end
```

```
function fugacity_slider()
% create figure & axes
 mainFig = figure('Name','Fugacity Demo','NumberTitle','off');
 ax = axes('Parent', mainFig, ...
 'Units','normalized', ...
 'Position',[0.12 0.30 0.75 0.65]);
% range of a_a
 a_a_min = 0.1;
 a_a_max = 0.45;
 a_a_init = 0.2;

% create the slider and position it at the bottom of the figure
 sld = uicontrol('Parent', mainFig, 'Style','slider', ...
 'Min',a_a_min, 'Max',a_a_max, 'Value',a_a_init, ...
 'Units','normalized', 'Position',[0.15 0.05 0.70 0.05], ...
 'Callback', @(src,~) redrawPlot(src.Value) );

% plot once at the initial slider value
 redrawPlot(a_a_init);

% function to redraw the plot as the value of a_a is changed

 function redrawPlot(a_a)
 % clear the current axes
 cla(ax);
 plot_fugacity_mixture(a_a, ax)
 title(ax, sprintf('Fugacity of a vs P (slider), a_a = %.2f', a_a));
 drawnow;
 end
end
```